REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

PLEASE DO NOT RETURN YOUR FORM TO THE ABOVE ORGANIZATION.

1. REPORT DATE *(DD-MM-YYYY)*	2. REPORT TYPE	3. DATES COVERED *(From - To)*
01-03-2010	FINAL	April 1st 2008- November 14, 2009

4. TITLE AND SUBTITLE
Bit-Error rate Monitor for Laser Communicatons

5a. CONTRACT NUMBER

5b. GRANT NUMBER
FA9550-08-1-0250

5c. PROGRAM ELEMENT NUMBER

6. AUTHOR(S)
ZRILIC, Djuro

5d. PROJECT NUMBER

5e. TASK NUMBER

5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
The University of New Mexico
Office of Research Administration, Scholes Hall Rm 102
Albuquerque, NM 87131-0001
(505) 277 8806

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)
OFC of Naval Rsch San Diego Regional Office
140 Sylvester Rd Bldg 140 RM 218
AFOSR/JA 875 N. randolph Street Rm 3112
Arlington, VA 22203 703 896 9500

10. SPONSOR/MONITOR'S ACRONYM(S)
ONRRO

11. SPONSOR/MONITOR'S REPORT NUMBER(S)
AFRL-SR-AR-TR-10-0157

12. DISTRIBUTION/AVAILABILITY STATEMENT
Public

13. SUPPLEMENTARY NOTES

20100426226

14. ABSTRACT

This equipment grant has made it possible to perform the research on award FA9950-07-1-0310. A semiconductor based laser source, fiber coupled to a modulator and fiber amplifier were purchased. The system was fiber coupled to a beam director, with the possibility to launch four beams in parallel to reduce scintillation noise. A range between 10 m to 22 km can be tested with this system. The equipment grant has made it possible to train our graduate students for the first time with modern digital oscilloscopes.

Real time BER estimator has been attempted using artificial neural network approach. After programming, the neural network outputs a continuous signal proportional to the bit error rate. This error level matched the reading of the bit error analyzer, only when the level of optical signal transmitted was kept at the same constant value used for the programming, and the whole optical link was bypassed. Considerable more parameters would have to be entered to the neural network program for a practical implementation to be possible.

15. SUBJECT TERMS
Line-of-sight communication, bit error rate,

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT	b. ABSTRACT	c. THIS PAGE			ZRILIC, Djuro
NC	NC	NC	none	30	19b. TELEPHONE NUMBER *(Include area code)* 505 454 3320

Bit-Error Rate Monitor for Laser Communications
DURIP - Final Performance Report
May 15th, 2008 - November 14th, 2009
Award No FA9950-08-1-0250

Dr. Djuro G. Zrilic. PI
Dr. Jean-Claude Diels and Dr. Alexandre Braga (Optics)
Center for High Technology Materials
The University of New Mexico
Albuquerque, New Mexico 87131
(505) 277-4026
e-mail: zrilic001@comcast.net;jcdiels@unm.edu

March 8, 2010

Contents

1 Introduction **2**

2 Electronics Part **3**
- 2.1 The Electronics Transmitter . . . 3
- 2.2 The Electronics Receiver . . . 3
- 2.3 The Pseudo-Error Detector Circuit . . . 4
- 2.4 The Microcontroller . . . 5
- 2.5 Software Implementation of NN . . . 6
- 2.6 Measurement Results . . . 8

3 BER Estimate for Adaptive Optics Control **10**

4 The Optical Link **10**
- 4.1 The Opto-electronic System . . . 11
- 4.2 The Launching System . . . 12
- 4.3 The Receiver System . . . 14

5 Concluding Remarks **17**

A BER Measurements **18**

B C Program **20**

Summary

An accurate estimate of bit-error rate (BER) in communication systems is essential to maintain accurate operation of the system. Currently BER estimate is assessed through off-line measurements which are typically carried out using bulky and expensive equipment. In addition to being a time consuming measurement method, it requires interruption of traffic, a problem not allowed in military communication links. To tackle this problem, we have implemented a test-bed experimental system for real time BER estimator using an artificial Neural Network (NN) approach which was trained off-line, programmed in C code and implemented using a PIC32 microcontroller. To create pseudo error conditions, without transmission of any additional data, an additional pseudo error optical path (at receiver) is created splitting the laser link into two separate paths. Measurement results of BER are in relatively close agreement with results obtained with dedicated BER equipment and results found in literature. The idea of on-line BER estimate can provide feedback for an adaptive control mechanism. For example, averaged BER signal (analog value) can be used for a pointing telescope control.

1 Introduction

The goal of this project has been to investigate the possibility of using on-line Bit Error Rate (BER) measurements in laser communication links. The main reasons for the investigation are:

- Existing BER measuring equipment is bulky and expensive.
- To measure BER with this equipment, communication traffic has to be interrupted.

We proposed to investigate the possibility of on-line BER measurement using an artificial Neural Network (NN) approach, where the BER unit estimator is an integral part of the receiver. As a result of this investigation, a BER monitoring unit was built using a pseudo error approach. Figure 1 presents a simplified block diagram of the experimental system. The transmitter consists of a digital

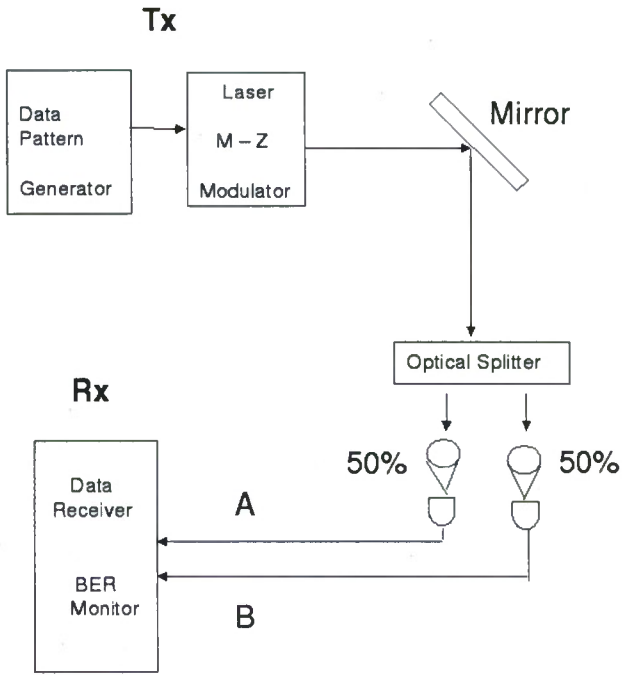


Figure 1: Block diagram of experimental BER monitor system.

pattern generator with scrambler circuit, a 1.5μm laser source with a Mach-Zender modulator, and a launching optics system, composed of collimating lenses and relay mirrors. The receiver consists of focusing optics elements, a 50/50 beam splitter, two photodetectors, a bit synchronizer circuit, a descrambler and a BER unit. A photograph of the experimental set up is shown in Fig. 2. This report describes the two essential parts of the system: electrical and optical. The first part of the report concerns the electrical part, and the second, the optical part. In part three, some recommendation and concluding remarks for further work are given, and part four (appendix) lists the program code and measurement table.

Figure 2: Experimental system set up.

2 Electronics Part

2.1 The Electronics Transmitter

The transmitting part of the experimental set up consists of pattern generator (HP 70841B), an interfacing circuitry, a 10 bit scrambler, and Mach-Zender Modulator (JDSU Uniphase Part No. 10020463-100). Figure 3 shows a simplified block diagram of the transmitter together with the transmitting optics.

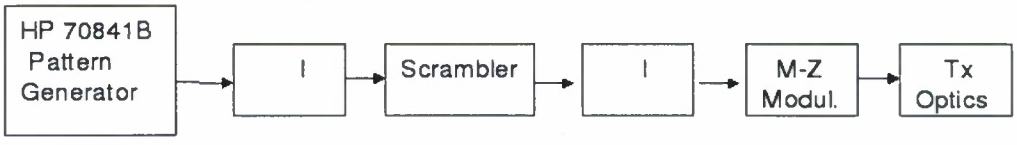


Figure 3: Simplified block diagram of transmitter.

2.2 The Electronics Receiver

A simplified block diagram of the receiver is shown in Fig. 4. It consists of an optimal and pseudo-receiver, a synchronizer, and a comparator. During propagation, the optical signal is exposed to different types of disturbances and distortion. To estimate the error of the received signal, the optical

link is divided into two paths: a main link A (no additional attenuation) and pseudo path (with some additional attenuation at the input of the receiver). The idea of using the parallel pseudo path is to estimate the BER in the main path, before it goes out of synchronization. For example, if the introduced attenuation of the pseudo path is two times higher than the attenuation of the main path, then the signal, in the pseudo path, is attenuated an extra 3 dB. This means that the pseudo path of the link will lose synchronization before the main path. Comparing synchronized data of the main link with a non-synchronized and additionally attenuated data of pseudo link, we can get an estimate of the BER in main optical link. With good statistics of the propagation conditions, an artificial Neural Network (NN) can be used as a BER estimator.

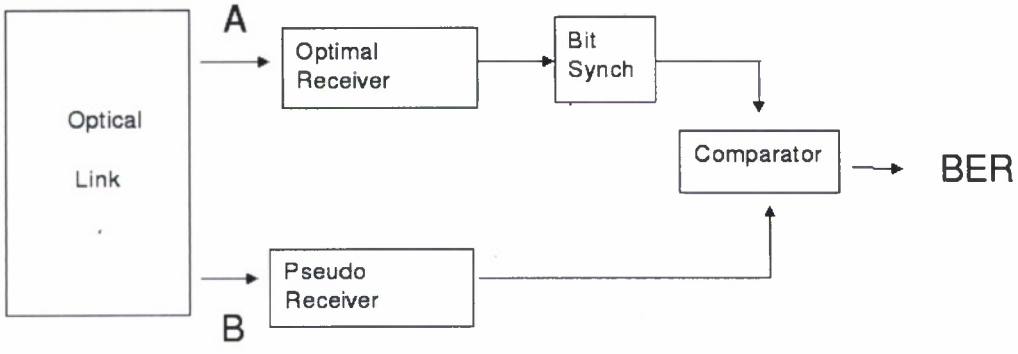


Figure 4: Simplified block diagram of receiver.

2.3 The Pseudo-Error Detector Circuit

To verify the pseudo error BER estimate idea, we have built the experimental set up shown in Fig. 5. Here, the main receiving path (A) consists of an optimal input filter, an attenuator A1, a threshold detector, an interface circuit, a bit synchronizer, a descrambler and the HP70842 B error performance analyzer. The pseudo receiving path (B) consists of a non-optimal input filter, an attenuator A2, a threshold detector, an error detector circuitry, and a PIC microcontroller with the NN. The error detector circuit consists of a delay line (for fine tuning), a digital comparator (XOR gate), a low pass filter, and an analog multiplier with amplifier.

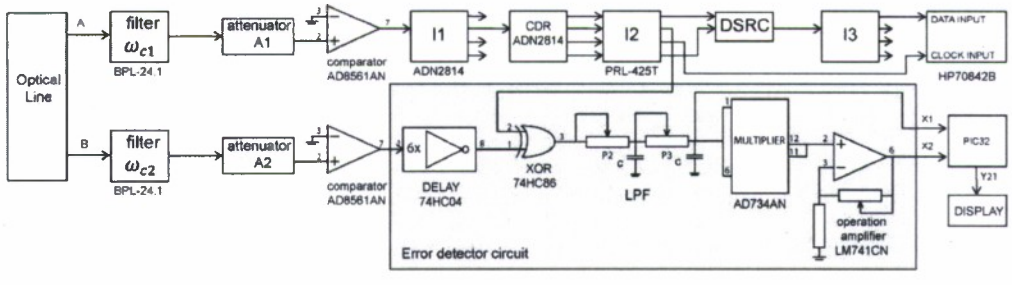


Figure 5: Block diagram of implemented system (Receiver).

The main path (A) is synchronized and evaluated in the HP BER analyzer. The BER data are recorded and evaluated for different levels of attenuation of the optical link. Synchronized data of

A link are compared with non-synchronized data of pseudo link B. In the case of perfect synchronization, A = B, the output of the digital comparator (XOR gate) is logical zero (thus BER = 0). As soon as the ideal propagation conditions change, there is a higher probability that the pseudo path will show more errors (because of additionally introduced attenuation), and the output of the XOR detector will indicate error pulses. For statistical proposes, this signal is averaged with a low pass filter and squared to get the variance. Both, X_1 and X_2 signals are fed as input to the programmable NN embedded in the PIC microcontroller. The complete schematic of the implemented error detector circuit is shown in Fig. 6.

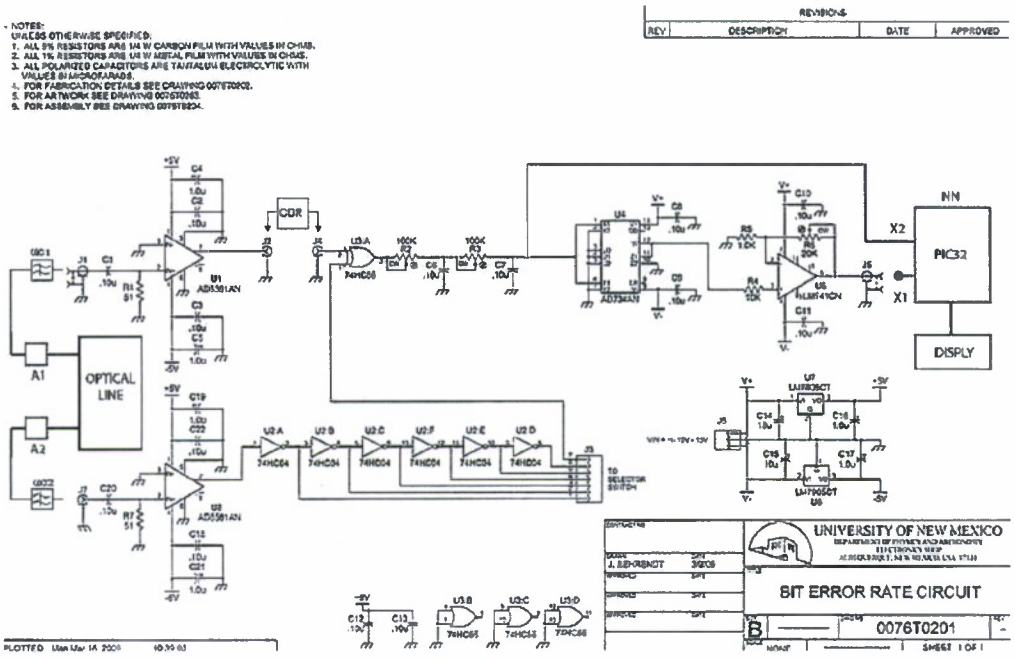


Figure 6: Schematic of a pseudo-error detector circuit.

2.4 The Microcontroller

Our previous experiments used Motorolas HCS12 microcontroller. Working with this controller was tedious and required a significant amount of time to become familiar with it. Our decision to switch to a PIC32 microcontroller was based on the following facts [1]:

- The PIC32 microcontroller kit contains everything needed to develop, program, debug and run codes as shown in a the block diagram of Figure 7.
- It is easy to interface with an USB cable.
- It is easy to learn how to program.
- It is low cost.

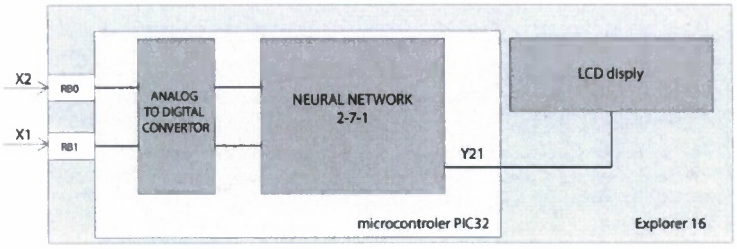


Figure 7: Block diagram of PIC32 microcontroller.

A block diagram of the PIC32 microcontroller is shown in Figure 7. The first step in the software implementation of the NN, using PIC32, was collection and evaluation of reliable BER data, taken with an HP BER meter setup (HP70841B and HP70842B).

In the time period of two weeks, we made a number of measurements, at room temperature, for various levels of attenuation of an optical link, in order to achieve good statistics. These data are averaged and classified in groups according to the level of a BER measured with the HP BER meter. We accepted 10^{-3} to be the highest acceptable level of BER, and 10^{-8} to be the lowest. The list of five classified groups of BER measurement is shown in Appendix A of this report. These measurements were taken at different levels of attenuation of the optical signal. Figure 8 shows the output of the error detector (XOR gate) when the level of the error is 10^{-8}.

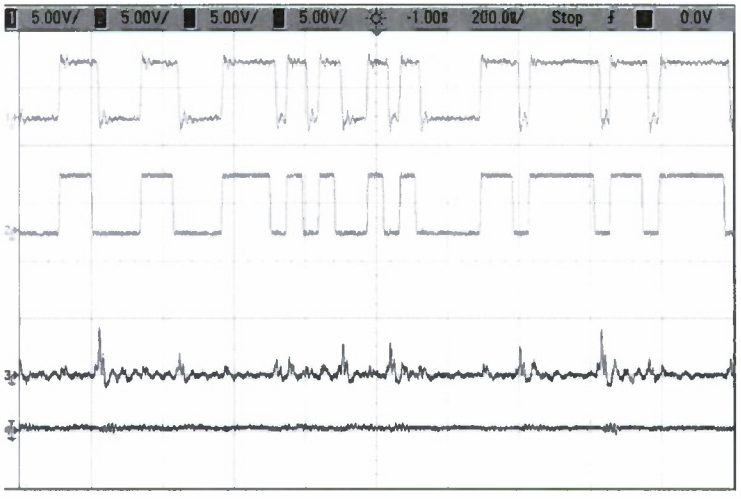


Figure 8: Output of the error detector in the presence of attenuation of an optical signal. Signal 1 input 1 in X-OR gate (see figure 5), signal 2 input 2 in X-OR gate. Signal 3 output of X-OR gate, signal 4 X2, DC input in controller, signal after RC filter

2.5 Software Implementation of NN

For our proposes, we chose a two-layer perceptron. In addition to the two inputs signals (X_1 and X_2 in Fig. 6), the perceptron has 7 hidden nodes (layer #1), and one output (layer # 2). Both layers have programming capabilities of weights and biases. Figure 9 shows the block diagram of the perceptron-

based NN used in our application, and Figure 10 shows a block diagram of a single neuron. Its output is calculated as $\mathbf{y = F(wx + b)}$.

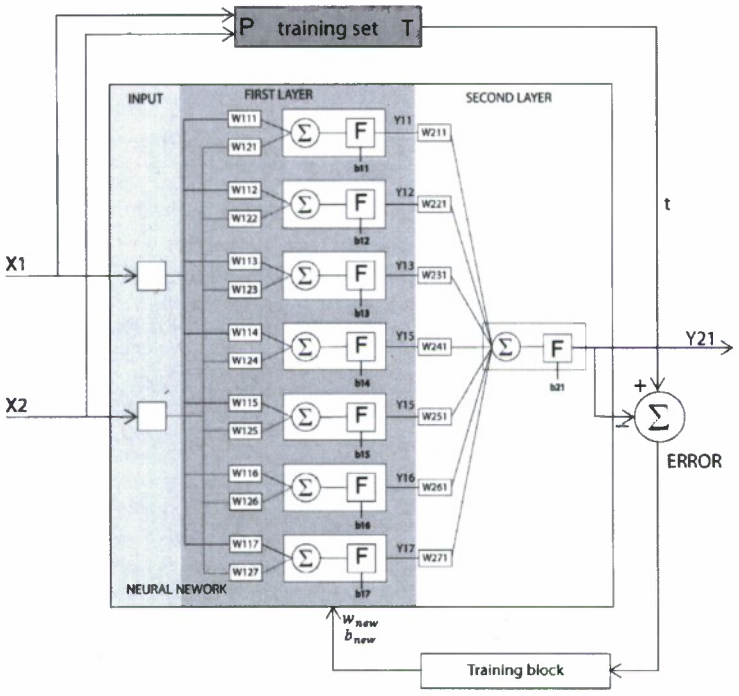


Figure 9: Block diagram of adaptive perceptron (Adeline).

The Transfer function of the perceptron adopted is of the form $\mathbf{y = 1/(1 + e^{-n})}$, and its graph is shown in Figure 11.

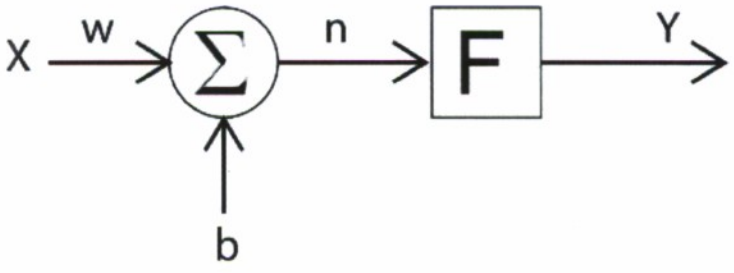


Figure 10: Single input neuron as a building block of the adaptive perceptron in Figure 9. X Input, W Weight, b Bias, F - Transfer function, Y Output.

Figure 9 consists of three blocks: a feed forward perceptron, a training block, and a training algorithm. The procedure to train the is as follows: first, it is necessary to form a training set. This set is formed in such a way that the values of input signals X_1 and X_2 are entered (stored) in memory location P. At the same time the BER is measured with an HP BER meter and this reading is recorded and entered into memory location T. After the training set is formed, the values of the memorized signals X_1 and X_2 are fed back to the inputs of the NN. For these values of input signals, the output of the NN, Y_21, is subtracted from the training signal t. If $Y_{21} = t$, then the error signal Error

= 0. This means that the coefficients and biases on the NN are already adjusted. In the case that error signals exist (positive or negative), the training block for adjustment of coefficients and biases generates signals w_{new} and b_{new} (for coefficients adjustment, see Figure 9). For every input signal, this procedure is repeated. In our case, measurements were made for six different levels of input signals X_1 and X_2 (in our experiment, maximum and minimum acceptable values of BER are 10^{-3} and 10^{-8} respectively). In Appendix A, the measured values are given of a BER at different levels

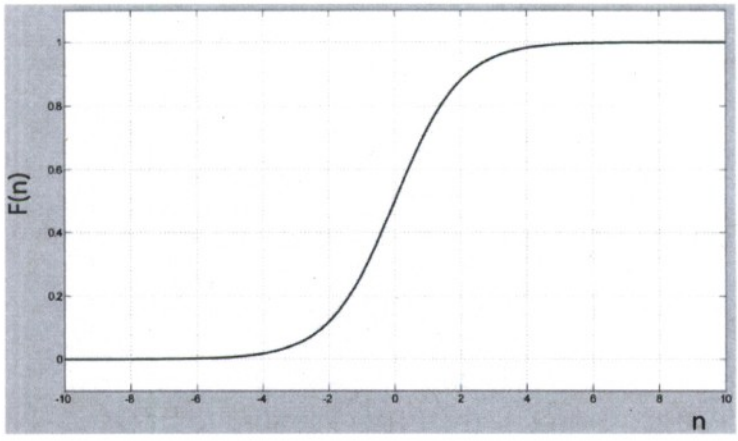


Figure 11: Sigmoid transfer function.

of input signals. Their mean values and respective variances are used as training data for the NN. A weight correction of layer coefficients is based on an algorithm proposed in reference [4], and its C code is enclosed in Appendix B.

2.6 Measurement Results

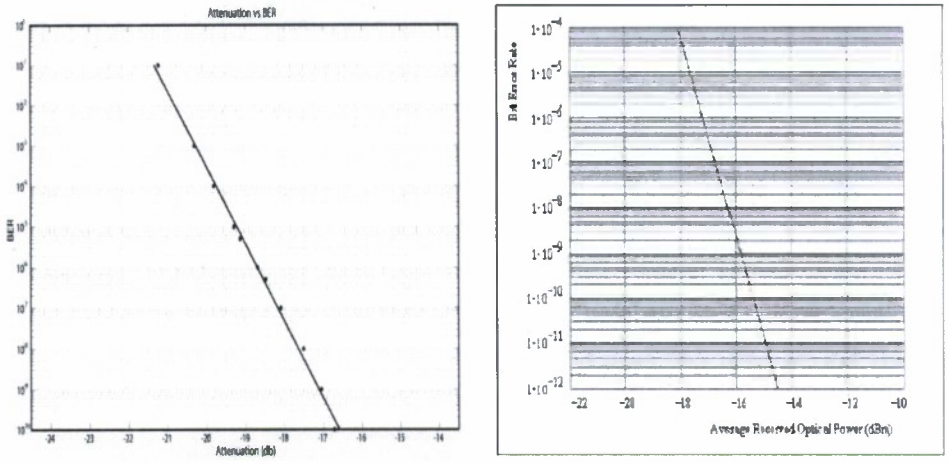
The system curve, BER as a function of attenuation level of the optical signal is shown in Figure 12. As can be seen, it is very steep and it is in general agreement with the result published in the literature [2].

All the most relevant equipment purchased for the electronics system are summarized as follows:

- **2.5GHz High Speed Oscilloscope:** in order to monitor the quality of the signal, a high performance, high speed, digital scope from Agilent (Infiniium Series No. DSO90254A) was used. It is crucial for the system to work properly that the received signal displays sharp edges, i.e transitions between the low values (logical '0's) and high values (logical '1's). That requires fast response equipment. For instance, to detect a signal with a rise time of 1 nsec, it would roughly require a system with a 1GHz bandwidth capability. This scope provided 2.5GHz bandwidth with 20Gsamples/sec sampling speed, which was more than adequate.

- **EZJIT jitter Analysis Software:** this software (DSO90000A-004) was part of the extended package purchased together with the Infiniium scope, and it is an essential tool for jitter signal analysis.

- **High Performance Probe:** a high performance, differential browse probe (E2675A), from Agilent, with a high impedance adapter (E2697A), and optionals (1131A,E2669A), was purchased



(a) (b)

Figure 12: (a) Results of measurement, and (b) Results from reference [2].

to assist with signal monitoring at points of the system where any disturbance of the node would be critical. One such a probe was necessary, for its capacity of "tapping" out a sample of the the signal without changing the characteristics of the information flow at that point on.

- **High Precision Delay Line:** to compensate for signal "de-phasing", one of the most fundamental requirements in the lab was a precision delay line. For that, a programmable delay line from Colby Instruments (HPDL-100A-10.23NS) was purchased. This delay line has the capacity to add delays in as little as 10psec increments within a 10nsec range.

- **Mixed Signal Scope:** on those points of the circuit where speed was not an issue, and which required constant monitoring, a low speed, low bandwidth, scope (MSO7034A) and probes (10074C), from Agilent, was used. This is a much rudimentary equipment compared to the (DSO90254A) scope, but nonetheless essential.

- **Free Space Broadband Detector:** several photodetectors were used in the project. For those specific measurements where bandwidth was a requirement, i.e sharp edge monitoring, a high bandwidth detector was necessary to match, in fact, surpass the bandwidth of the fast scope (DSO90254A). For that, a 25GHz broadband photodetector from NewFocus (1437) was purchased.

- **Power Supplies:** several high quality power supplies were necessary to bias all the electronic circuitry, at several voltage and power levels. The best choice was to use power supplies from Agilent (E3616A,E3620A,E3610A).

- **Buffers and Voltage Translators:** to assist with signal distribution, regeneration, and voltage level translation, required for interfacing electronic circuits of different voltage standards, several 50ohm line drivers (fannout buffers) and translators, from Pulse Research Lab (PRL-420ND,PRL-450ND, PRL-414B) were purchased.

3 BER Estimate for Adaptive Optics Control

Laser communication systems, such as in airborn and submarine platforms, use line of sight optical links. In such links it is important that communication link remains invisible. Link between platforms must be reliable in the presence of a variety of disturbances such as atmospheric and hidrospheric perturbations, movements, bums, etc. To overcome this obstacles, a reliable control of communication link must be achieved. In our opinion, BER estimate of the laser link can be used for this goal. Figure 13 presents a simple block diagram of the proposed control system based on BER estimate.

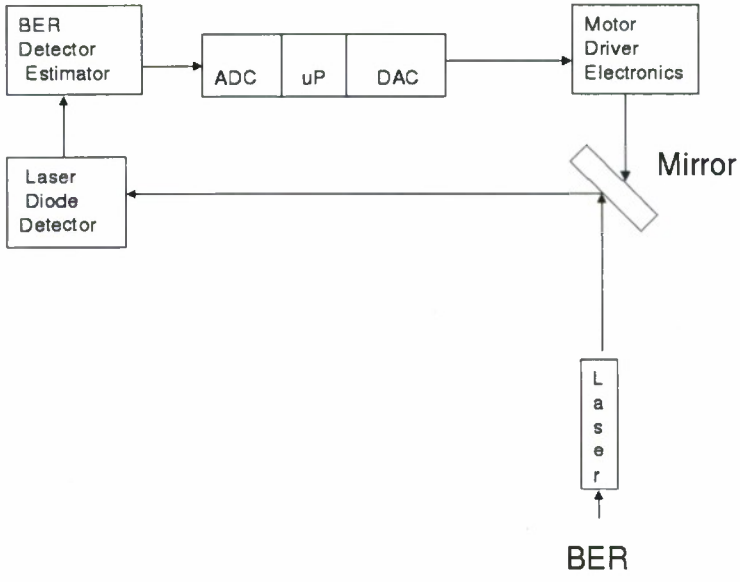


Figure 13: Block diagram of proposed control system.

A BER estimate is detected in point 3 (the output of the XOR gate in Figure 5) and is modulated by a laser modulator, then reflected by a mirror and detected by four quadrant diodes. Quadrant diode detectors consists of 4 photo sensitive diodes and the outputs produce voltages proportional to the amount of photon flux. When a laser beam is focused in the middle of the detector, then every diode has one quarter of its light, and the voltage levels at the outputs of the diodes are equal. As the beam moves off-center, the proportion of light on each diode changes. As a consequence, their output potentials change accordingly. These changes in voltages are detected by BER electronics (digital BER signal is averaged to obtain a nearly DC level signal) and processed in micro-controller. The differences in the diodes potentials determine the magnitude and location of error in the beam spot position. Beam steering can be accomplished with a mirror mounted on 2 DC motors.

4 The Optical Link

The optical link is the part of the system where the information was carried optically using a laser beam. It relates to the generation, modulation, propagation and detection of the optical signal. During the project, several changes in the optics were made to enhance the performance of the link. The following sections summarize the evolution of the system from one year to the next.

In the first year, the most basic optical link was assembled. It was a single channel system, where the optical signal was propagated all the way up to the roof of the CHTM building facility and back. Figure 14 shows the schematics of the link.

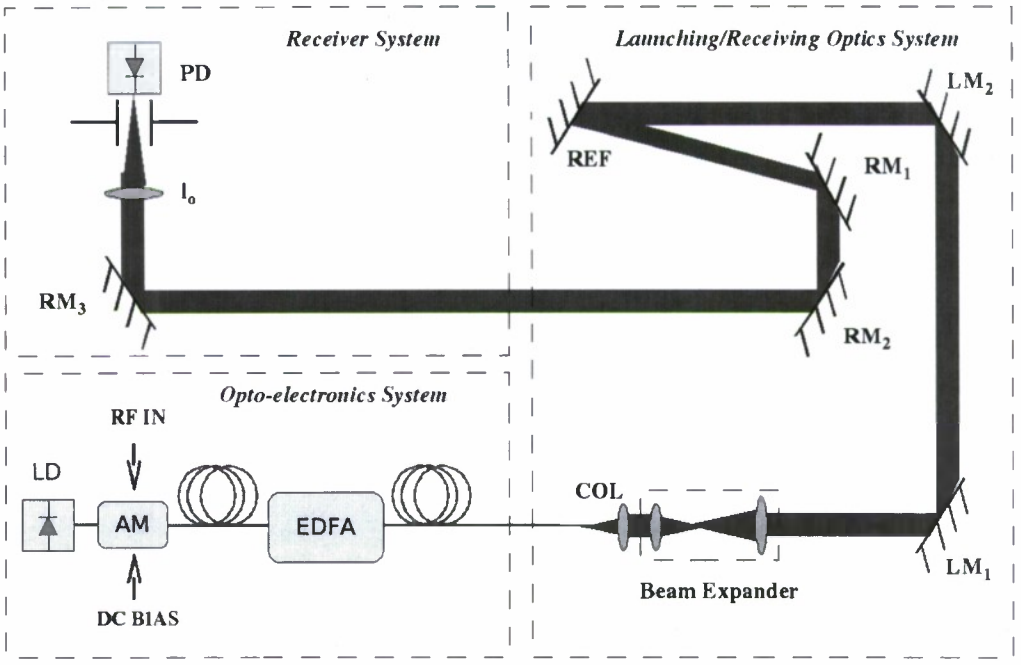


Figure 14: Schematics of the basic optical link system implemented in FY#1: opto-electronics, launching/receiving optics and receiver system. Blue lines indicate optical fiber paths. Red lines indicate light in air.

The system is divided in three separate sections: (1) the opto-electronics system, where the optical signal is generated, modulated, and amplified, (2) the launching system, where the amplified modulated light is propagated in air and directed, through lenses and relay mirrors to the third section, and (3) the receiver system, where it is collected and imagined onto a photodetector (PD) to be transformed back into an RF modulated signal.

4.1 The Opto-electronic System

In the opto-electronics system, a Laser Diode (LD) light source is directly connected to an optical Amplitude Modulator (AM), which is interfaced to the electronics of the system via an RF input signal port. The output of the modulator goes to an Erbium Doped Fiber Amplifier (EDFA), for amplification using standard Single Mode Fibers (SMF). The most important components purchased for this system are listed below:

- **The Laser Diode (LD) Source:** a PM fiber coupled, 20mW, 1550nm center wavelength, CW DFB laser diode, from JDSU Uniphase (CQF935/508 Series). The LD was driven using a Spectra Diode Labs laser diode driver (SDL-800), and operated at an average output power of 15mW.

- **The Amplitude Modulator (AM):** a 10Gb/sec, "bias-ready" intensity modulator from JDSU Uniphase. The modulator was coupled with PM fiber at the input, and directly attached to the laser source via an FC/UPC mating sleeve connector. The output fiber was standard SMF.

- **The Optical Amplifier:** an Erbium-Doped Fiber Amplifier (EDFA) from IPG Photonics (Model EAR-5K-C). The amplifier operates with a minimum input power of -5dBm, with was suitable to the low power output level coming out of the modulator.

4.2 The Launching System

In the launching system, or emitter, the light coming from the EDFA is collimated using a fiber ready, FC female type collimator (COL), followed by a beam expander. From this point on, a series of metal film, flat mirrors (LM_1, LM_2) relayed the 5cm diameter optical beam through the buildings roof, back into the optical table (REF, RM_1, RM_2, RM_3), where a single biconvex lens (l_o) focused the light into an InGaAs type photodetector (PD). In the second year, an improved system for the emitter was implemented. In comparison to the system used in the previous year, this new system offered the following advantages:

- **Splitting of the emitted optical beam:** as mentioned in the original proposal, splitting the laser into multiple beams enhances the performance of the system, as it helps minimize the effects of scintillation [3, 5, 6], which results from propagation in the atmosphere. In this new system, the beam was split into 4 separate channels using a high power fiber optic coupler.

- **Attachment of the emitter to the beam director:** an important innovation of the new optical system is the assembly of the whole launching optics on the frame of the beam director located on the buildings roof. In this way, the entire launching system was attached to the frame of rotation of the beam director, and alignment of the emitter was no longer necessary every time the beam director was rotated.

- **Use of larger aperture launching mirrors:** Compared to the 3 diameter mirrors used in the previous year, new 6 diameter flat mirrors allowed the possibility of placing the target at much farther distances, as it can be used to increase the Rayleigh range of the propagating beam .

Schematics of the new emitter system are shown in Figure 15. It starts with the amplified optical signal, which comes out of the Erbium-doped fiber amplifier. The collimated beam is coupled to a section of Single-Mode Fiber (SMF) using an XYZ fiber launching system (Thorlabs, Model No.MBT612), as seen in Figure 16. The other end of the fiber is connected to a high power 1 x 4 optical coupler (NEPTEC No. CC-FA14BO0419E) that splits out the beam into 4 separate channels, where the light propagates along 4 separate sections of SM fibers. The output fibers are taken directly to the roof, inside the beam director (Fig. 20). There, they are mounted on 90 degree racks, together with 15cm focal length lenses to form a collimating assembly (see inlet in Fig. 17). The racks are placed right below 4 6" metal flat mirrors which are then used to relay the beams to the reflecting target (see REF in Fig. 14).

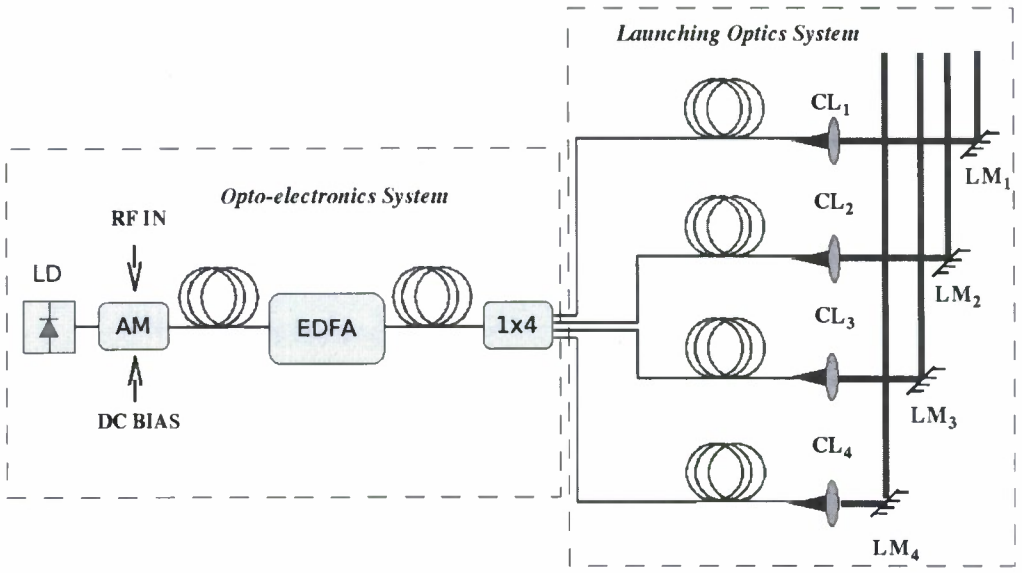


Figure 15: Schematics of the improved emitter system: opto-electronics and launching optics. Blue lines indicate optical fiber paths. Red lines indicate light in air.

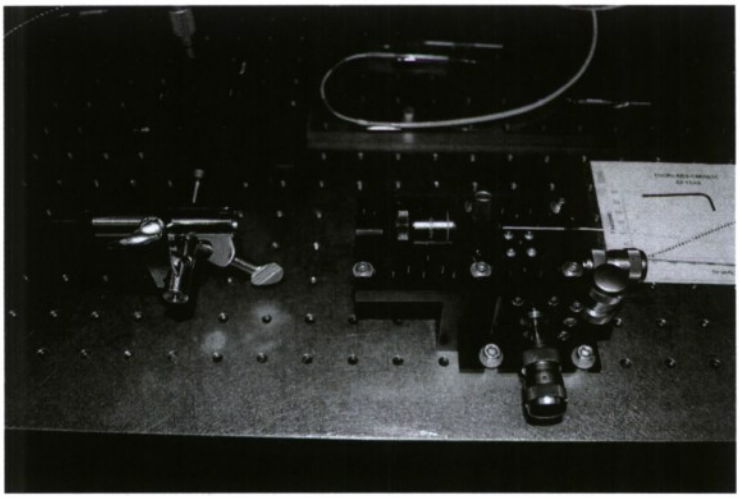


Figure 16: Fiber launching system, where a collimated amplified beam (left) is coupled into a SM fiber using a microscope objective mounted on a 3 axis positioning stage (right).

All the most relevant parts purchased for this system are summarized as follows:

- **Launching Mirrors (LM_{1-4}):** 4 6"dia. fused silica, protected silver, flat mirrors (CVI Custom Specified) were at the final stage of the emitter, launching the collimated beams to the reflecting mirror (REF) which pointed them toward the receiver's mirror.
- **Beam Splitter (BS):** a 1x4 high power beam splitter, coupled to the output of the EDFA, split the beam into 4 separate, equal power channels. The beam splitter used was from NEPTEC Optical Solutions (P/N CC-SA14B0Q419).

- **Optical Fiber Patchcords:** 4 8 meters long optical fiber patchcords carried the signal all the way up to the building's roof, where the beam-director was placed. The patchcords were SMF standard, connected to the ports of the BS via FS/APC mating sleeve connectors. The APC types of connectors come with an angle at the fiber end to minimize reflections back into the system, which would be damaging at those power levels.

- **Collimators (CL_{1-4}):** four 2" dia. IR CaF_2 singlet spherical plano-convex lenses (CVI PLCX-50.8-83.8-CFIR-1550) placed at the collimation stages directly above the fiber ends, inside the beam director, provided collimation to the output beams in air.

4.3 The Receiver System

In the third, and final year of the project, an improvement was made on the Receiver system. Figure 18 shows the schematics of the system. In comparison to the system of the previous year, this new system provided:

- **Addition of a second channel to the received signal:** using a polarization independent 50/50 beam splitter, the received signal was split into two beams, allowing monitoring of the signals at two separate channels, as required by the electronics system.

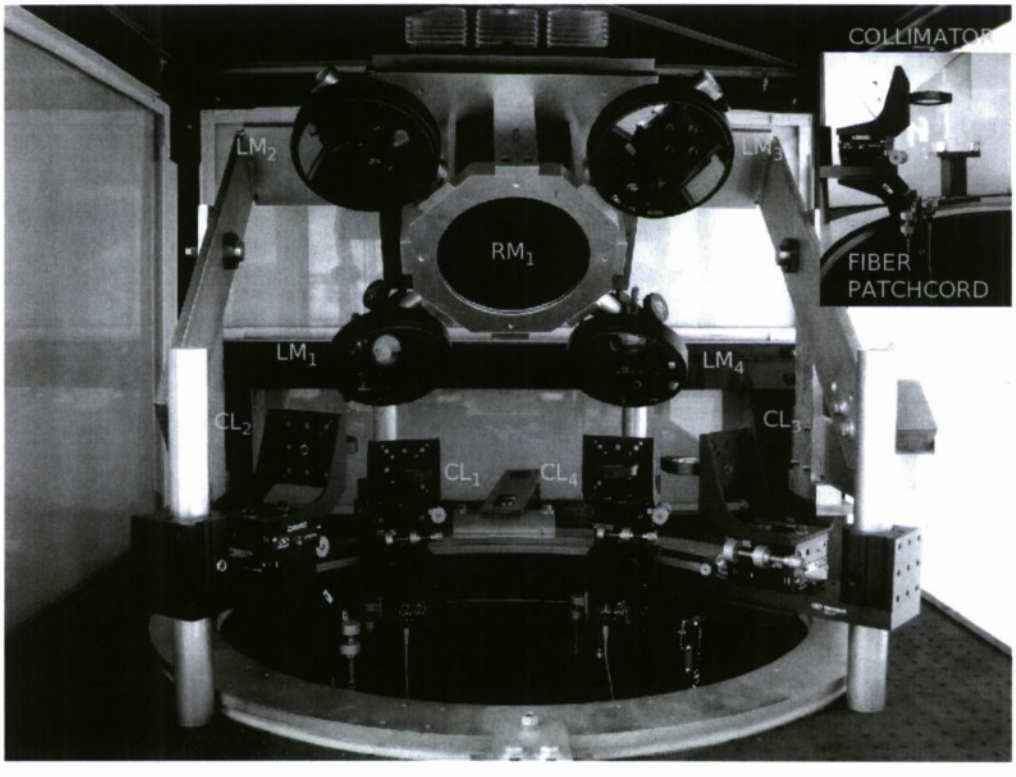


Figure 17: Photograph of the entire emitter system: the launching optics inside the beam director (background), and detail of the fiber collimator assembly (inlet).

- **Insertion of a linear attenuator:** an attenuator was placed before the beam splitter to allow gradual linear attenuation of the received signal.

- **High performance photodetectors:** to equalize the response of two channels of the receiving system, two new high performance photodetectors were placed at the end of the link. These new detectors, with bigger active areas, improved focusing, and, therefore, signal detection.

Figure 18 shows the schematics of the new receiver. It consists of 5 mirrors (RM_1, and M_{1-5}), 1 attenuator, 1 50/50 beam splitter (BS), two focusing lenses (l_{o1-2}), and 2 photodetectors (PD_{1-2}). The first (and largest) mirror is mounted at the beam director, together with the launching system mirrors (the large mirror at the center of Fig. 17). It relays the four incoming beams to a mirror placed directly below it, inside the lab (M_1), at the optical bench (same as RM_2 in Fig. 17). This mirror relays the beam to M_2, which, in turn, relays to M_3. An attenuator, placed right in front of M_4, allows gradual linear attenuation of the optical signal, which is then sent to a 50/50, polarization insensitive, beam splitter (BS). At this point, half of the signal is focused to one photodetector (PD_1) via l_{o1}, whereas the other half is focused to PD_2 via l_{o2}, after passing another relay mirror M_5.

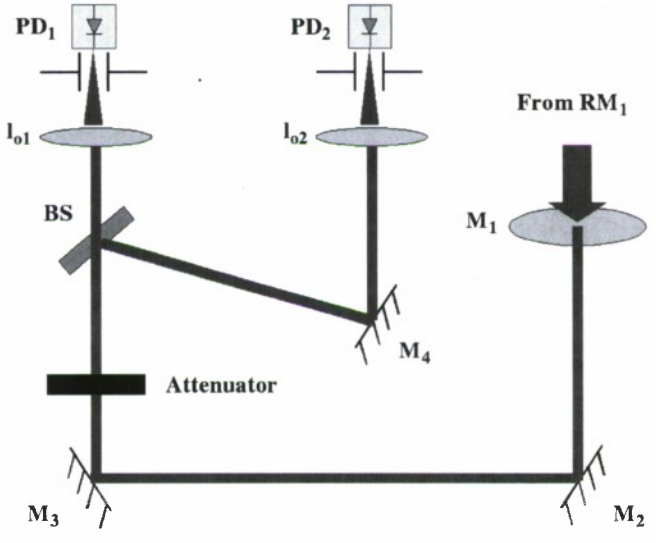


Figure 18: Schematics of the receiver system. Red lines indicate light in air.

The new purchased parts which were added to the system are summarized as follows:

- **50/ 50 Beam Splitter (BS):** a 2"dia. 50/50 beam splitter from CVI laser (Part. No. BSNP-1550-50-2025).

- **Linear Attenuator:** a mounted linear, variable, metallic ND filter (Thorlabs, Inc. Part No. NDC-100-4M).

- **Focusing lenses (l_{o1-2}):** 2 2" dia., 50cm focal length, biconvex lenses (CVI laser, Part. No. BICX-59.8-49-3-C,108-10,33-20).

- **Photodetectors (PD_{1-2}):** 2 0.2mm^2 active area, wideband amplified InGaAs detectors (Thorlabs, Inc. Part. No. PDA10CF).

Figure 19 shows a TOP VIEW picture of the actual new receiver system. The red lines indicates the trajectory of the optical signal along the elements of the receiver.

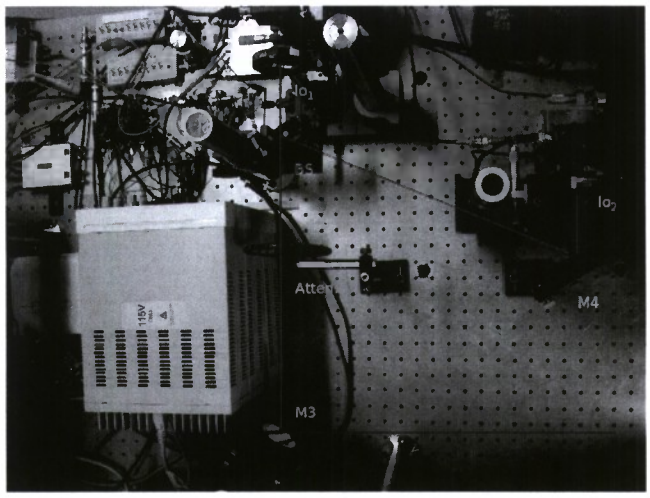


Figure 19: Photograph of the actual receiver system. Red lines indicate the light path in air.

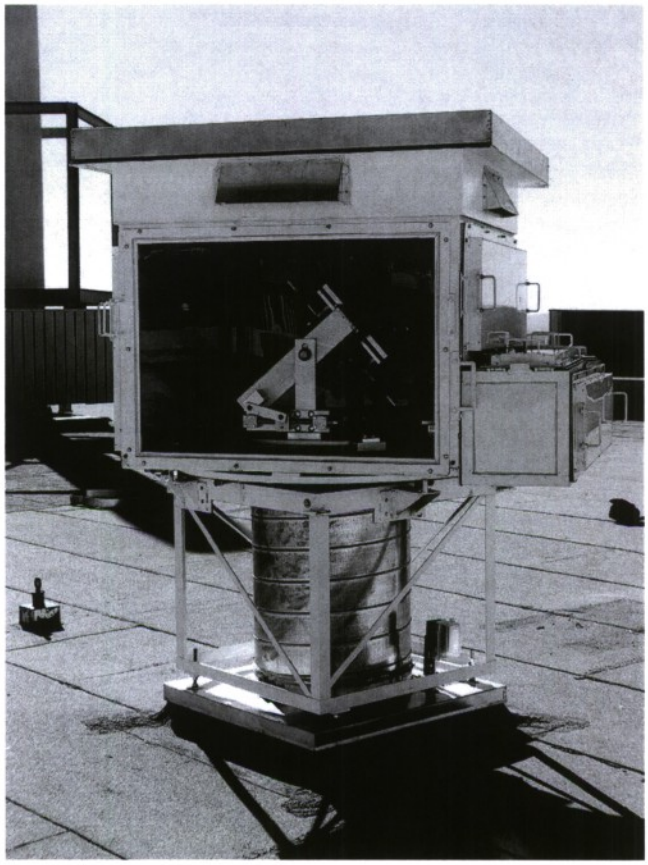


Figure 20: Photograph of the beam director located at the roof of the CHTM facility.

5 Concluding Remarks

Our experimental results have shown that real time BER monitoring system has potential applications in different areas of digital communication links. In addition, being integral part of the receiver, signal of BER estimator can be used for control purposes. Our additional goal was to implement NN system with on-line training capabilities. Unfortunately, I, Dr Zrilic, have faced permanent problem of keeping qualified graduate students. In past three years six different graduate students - Alex P. Braga, Pavlov, Behshad Amadi, Zhuoyao Wang, Kai Du, Mladen Prijic - were employed.

APPENDIX

A BER Measurements

	X_2 (mV)	X_1 (mV)	Error	X_2 (mV)	X_1 (mV)	Error	X_2 (mV)	X_1 (mV)	Error
	87	42	10^{-8}	103	84	10^{-6}	115	95	10^{-4}
	62	34	10^{-8}	103	76	10^{-6}	113	96	10^{-4}
	71	29	10^{-8}	93	66	10^{-6}	121	96	10^{-4}
	66	41	10^{-8}	110	76	10^{-6}	115	98	10^{-4}
	65	40	10^{-8}	107	71	10^{-6}	116	88	10^{-4}
	79	32	10^{-8}	110	77	10^{-6}	114	92	10^{-4}
	71	34	10^{-8}	101	59	10^{-6}	124	95	10^{-4}
	69	27	10^{-8}	96	73	10^{-6}	112	97	10^{-4}
	73	39	10^{-8}	93	62	10^{-6}	117	86	10^{-4}
	69	27	10^{-8}	92	60	10^{-6}	116	104	10^{-4}
	73	43	10^{-8}	101	63	10^{-6}	126	93	10^{-4}
	69	34	10^{-8}	107	77	10^{-6}	118	91	10^{-4}
	70	40	10^{-8}	99	66	10^{-6}	122	96	10^{-4}
	82	29	10^{-8}	104	73	10^{-6}	117	99	10^{-4}
	69	36	10^{-8}	100	64	10^{-6}	69	100	10^{-4}
	72	42	10^{-8}	97	61	10^{-6}	72	101	10^{-4}
	83	39	10^{-8}	109	65	10^{-6}	83	98	10^{-4}
	73	36	10^{-8}	101	75	10^{-6}	73	87	10^{-4}
	75	43	10^{-8}	104	71	10^{-6}	75	89	10^{-4}
	85	38	10^{-8}	98	79	10^{-6}	85	90	10^{-4}
mean	73.15	36.25		101.4	69.9		105.15	94.55	

Table 1: BER Measurements.

	X_2 (mV)	X_1 (mV)	Error	X_2 (mV)	X_1 (mV)	Error	X_2 (mV)	X_1 (mV)	Error
	100	67	10^{-7}	113	82	10^{-5}	154	140	10^{-3}
	104	56	10^{-7}	121	94	10^{-5}	152	151	10^{-3}
	102	56	10^{-7}	120	89	10^{-5}	160	145	10^{-3}
	87	67	10^{-7}	108	82	10^{-5}	159	146	10^{-3}
	90	60	10^{-7}	118	77	10^{-5}	152	147	10^{-3}
	87	58	10^{-7}	109	97	10^{-5}	150	150	10^{-3}
	100	57	10^{-7}	106	86	10^{-5}	148	155	10^{-3}
	93	66	10^{-7}	103	90	10^{-5}	149	153	10^{-3}
	84	61	10^{-7}	112	91	10^{-5}	150	151	10^{-3}
	90	50	10^{-7}	117	78	10^{-5}	147	149	10^{-3}
	89	62	10^{-7}	123	80	10^{-5}	156	143	10^{-3}
	90	64	10^{-7}	105	79	10^{-5}	144	141	10^{-3}
	98	51	10^{-7}	112	93	10^{-5}	148	139	10^{-3}
	87	50	10^{-7}	119	87	10^{-5}	151	143	10^{-3}
	84	49	10^{-7}	104	94	10^{-5}	153	140	10^{-3}
	92	54	10^{-7}	117	90	10^{-5}	154	141	10^{-3}
	90	64	10^{-7}	122	82	10^{-5}	161	147	10^{-3}
	98	61	10^{-7}	115	89	10^{-5}	156	148	10^{-3}
	85	63	10^{-7}	113	87	10^{-5}	150	148	10^{-3}
	91	59	10^{-7}	115	79	10^{-5}	151	145	10^{-3}
mean	92.05	58.75		113.6	86.3		152.25	146	

Table 2: BER Measurements.

B C Program

```
/*
** main.c
**
*/

// configuration bit settings, Fcy=72 MHz, Fpb=36 MHz
#pragma config POSCMOD=XT, FNOSC=PRIPLL
//New Oscillator Selection bits
#pragma config FPLLIDIV=DIV_2, FPLLMUL=MUL_18, FPLLODIV=DIV_1
//Pll input driver, Pll multiplier, Pll output driver
#pragma config FPBDIV=DIV_2, FWDTEN=OFF, CP=OFF, BWP=OFF
//Peripheral Bus Clock Divisor, WatchDog Timer Enable bit,
 Code-Protect bit, Code-Protect bit

#include <p32xxxx.h>
#include "Explore.h"
#include "ADC.h"
#include "LCD.h"
#include "NeuralNetwork.h"

main (){
  int read, s, u, v, samp;
  float input0, input1;
  struct neuralnetwork x;

  initADC(AINPUTS); // initialize the ADC
  initLCD();  // initialize the LCD display

  x = inicialization(x);
  x = inicializationWeight(x);
  x = trainNN(x);

  while(1){
    input0=ADC(0);
input1=ADC(1);
x=putInputs(x, input0, input1);
x=propagation(x);
Delayms(2000);
displayFloat(x.layers[2].output[0]);
  }  // main loop
}

/*
```

```
** LCDlib.c
**
*/

#include <p32xxxx.h>
#include <plib.h>
#include "Explore.h"
#include "LCD.h"

#define PMDATA PMDIN

void initLCD( void)
{
// PMP initialization
mPMPOpen(PMP_ON | PMP_READ_WRITE_EN | 3,
PMP_DATA_BUS_8 | PMP_MODE_MASTER1 |
PMP_WAIT_BEG_4 | PMP_WAIT_MID_15 |
PMP_WAIT_END_4,
0x0001,  // only PMA0 enabled
PMP_INT_OFF); // no interrupts used
// wait for >30 ms
Delayms(30);
//initiate the HD44780 display 8-bit init sequence
PMPSetAddress(LCDCMD);  // select command register
PMPMasterWrite(0x38); // 8-bit int, 2 lines, 5x7
Delayms(1);  //>48 us
PMPMasterWrite(0x0c); // ON, no cursor, no blink
Delayms(1);  //>48 us
PMPMasterWrite(0x01); // clear display
Delayms(2);  //>1.6 ms
PMPMasterWrite(0x06); // increment cursor, no shift
Delayms(2);  //>1.6 ms
}  // initLCD

char readLCD( int addr) // readLCD
{
PMPSetAddress(addr); // select register
mPMPMasterReadByte(); // initiate read sequence
return mPMPMasterReadByte(); // read actual data
}

void writeLCD( int addr, char c)
{
while(busyLCD());
PMPSetAddress(addr);
```

```
PMPMasterWrite(c);
}
void putsLCD(char *s)
{
while(*s)
putLCD(*s++);
}

void displayInt(unsigned int integer)
{
if(integer/10000)
putLCD(0x30 + (integer / 10000));
if(integer/1000)
putLCD(0x30 + (integer % 10000) / 1000);
if(integer/100)
putLCD(0x30 + ((integer % 10000) % 1000) / 100);
if(integer/10)
putLCD(0x30 + (((integer % 10000) % 1000) % 100) / 10 );
putLCD(0x30 + (((integer % 10000) % 1000) % 100) % 10 );
}

void displayFloat(float number)
{
int j,i;
float remainder;
j=0;
clrLCD();
putsLCD( "Error ratio" );
cmdLCD( 0x80 | 0x40);
if (number>0)
{
while (number<1)
{
j=j+1;
number=number*10;
}
if (j>25)
{
number=0.0;
displayInt((unsigned int)number);
putLCD(0x2e);
remainder=(number - ((unsigned int)number))*10;
displayInt((unsigned int)remainder);
remainder=(remainder - ((unsigned int)remainder))*10;
displayInt((unsigned int)remainder);
```

```
putsLCD("x10-0");
displayInt(0);
}
else
{
displayInt((unsigned int)number);
     putLCD(0x2e);
remainder=(number - ((unsigned int)number))*10;
displayInt((unsigned int)remainder);
remainder=(remainder - ((unsigned int)remainder))*10;
displayInt((unsigned int)remainder);
putsLCD("x10-0");
displayInt(j);
}
}
// write negative number
else
{
putLCD(0x2d);
number=-number;
while (number<1)
{
j=j+1;
number=number*10;
}
displayInt((unsigned int)number);
    putLCD(0x2e);
remainder=(number - ((unsigned int)number))*10;
displayInt((unsigned int)remainder);
remainder=(remainder - ((unsigned int)remainder))*10;
displayInt((unsigned int)remainder);
putsLCD("x10-0");
displayInt(j);
}
//Delayms(2000);
}

/*
** ADClib.c
**
*/

#include <p32xxxx.h>
#include <plib.h>
#include "Explore.h"
```

```
#include "ADC.h"


void initADC(int amask){
AD1PCFG=amask;
AD1CON1=0x00E0;
AD1CSSL=0;
AD1CON2=0;
AD1CON3=0x1F3F;
AD1CON1SET=0x8000;
}

int readADC(int ch){
AD1CHSbits.CH0SA=ch;
AD1CON1bits.SAMP=1;
while(!AD1CON1bits.DONE);
return ADC1BUF0;
}

float ADC (int port){
int i ;
float average, sweep;
sweep=0;
for(i=0; i<20; i++){
sweep += readADC(port);
Delayms(300);
}
average=sweep/20;
return average;
}

/*
** Explore.c
**
*/

#include <p32xxxx.h>
#include <plib.h>
#include "explore.h"

void initEX16( void){
#ifndef PIC32_STARTER_KIT
mJTAGPortEnable(0);
#endif
SYSTEMConfigPerformance( FCY);
```

```
INTEnableSystemMultiVectoredInt();
LATA = 0;
TRISA = 0xFF80;
}

void _general_exception_handler(unsigned c, unsigned s){
while (1);
}

/*
** Simple Delay functions
*/

void Delayms(unsigned t){
T1CON = 0x8000;
while (t--)
{
TMR1 = 0;
while (TMR1 < FPB/1000);
}
}

/*
** NeuralNetwork.c
**
*/

#include <p32xxxx.h>
#include <plib.h>
#include "explore.h"
#include "LCD.h"
#include "NeuralNetwork.h"

struct neuralnetwork inicialization(struct neuralnetwork x){

  x.layers[0].n=2;
  x.layers[1].n=7;
  x.layers[2].n=1;

  x.trainSet[0][0][0]=73;
  x.trainSet[0][0][1]=36;
  x.trainSet[0][1][0]=0.000000015;
  x.trainSet[1][0][0]=92;
  x.trainSet[1][0][1]=59;
  x.trainSet[1][1][0]=0.00000015;
```

```
  x.trainSet[2][0][0]=101;
  x.trainSet[2][0][1]=70;
  x.trainSet[2][1][0]=0.0000015;

  x.trainSet[3][0][0]=114;
  x.trainSet[3][0][1]=86;
  x.trainSet[3][1][0]=0.000015;

  x.trainSet[4][0][0]=118;
  x.trainSet[4][0][1]=94;
  x.trainSet[4][1][0]=0.00015;

  x.trainSet[5][0][0]=132;
  x.trainSet[5][0][1]=110;
  x.trainSet[5][1][0]=0.0015;

  return x;
}
struct neuralnetwork inicializationWeight(struct neuralnetwork x){
  int s, u, v;
  // initialization weight in layer 0
s=0;
for(u=0; u<x.layers[s].n; u++)
for(v=0; v<x.layers[s+1].n-1; v++){
x.weight[s][0][v]=0.01901;
x.weight[s][1][v]=0.01101;
}
  //initialization weight in layer 1
s=1;
for(u=0; u<x.layers[s].n; u++)
for(v=0; v<x.layers[s+1].n; v++){
        x.weight[s][u][v]=7.201;
}
  //initialization bias in first layer
for(u=0; u<x.layers[s].n; u++){
x.layers[s].bias[u]=-4.2501;
}

  //initialization bias in second layer
x.layers[2].bias[0]=-22.301;

  // initialization delta weight
for(s=0; s<NUM_LAYERS-1; s++)
    for(v=0; v<x.layers[s+1].n; v++)
```

```
      for(u=0; u<x.layers[s].n; u++) {
x.weightP[s][u][v]=0.0;
}

  // initialization delta bias
for(s=1; s<NUM_LAYERS-1; s++)
     for(v=0; v<x.layers[s].n; v++) {
x.layers[s+1].biasp[v]=0.0;
}
  return x;
}

float sigmoid(float net) {
  return 1/(1+exp(-net));
}

struct neuralnetwork propagation(struct neuralnetwork x) {
  int s, v, u;
  float net = 0.0;
  for(s=1; s<NUM_LAYERS; s++)
   for(v=0; v<x.layers[s].n; v++) {
     net = x.layers[s].bias[v];
       for(u=0; u<x.layers[s-1].n; u++)
        net += x.layers[s-1].output[u]*x.weight[s-1][u][v];
       x.layers[s].output[v] = sigmoid(net);
     }
  return x;
}

struct neuralnetwork putInputs(struct neuralnetwork x,
 float input1, float input2){
  x.layers[0].output[0] = input1;
  x.layers[0].output[1] = input2;
  return x;
}

struct neuralnetwork putInputsTrainSet(struct
neuralnetwork x, int set){
  int u;
  for(u=0; u<x.layers[0].n; u++)
    x.layers[0].output[u] = x.trainSet[set][0][u];
  return x;
}

struct neuralnetwork error (struct neuralnetwork x, int set) {
```

```
  int v, u, s;
  float f, sigmaa;
  // error in output (second) layer
for( v=0; v<x.layers[NUM_LAYERS-1].n; v++){
     x.layers[NUM_LAYERS-1].difference[v] =
x.trainSet[set][1][v] - x.layers[NUM_LAYERS-1].output[v];
     x.error += (x.layers[NUM_LAYERS-1].difference[v])
*(x.layers[NUM_LAYERS-1].difference[v]);
        x.layers[NUM_LAYERS-1].sigma[v]=x.layers[
NUM_LAYERS-1].difference[v]*x.layers[NUM_LAYERS-1].output[v]*
(1 - x.layers[NUM_LAYERS-1].output[v]);
   }

  // error in first layer
for(s=NUM_LAYERS-2; s>=0; s--){
     for(u=0; u<x.layers[s].n; u++) {
sigmaa = 0.0;
for(v=0; v<x.layers[s+1].n; v++){
sigmaa += x.layers[s+1].sigma[v]*x.weight[s][u][v];
}
f = x.layers[s].output[u];
x.layers[s].sigma[u]= f*(1-f)*sigmaa;
}
}
  return x;
}

struct neuralnetwork trainNN(struct neuralnetwork x)
{
int iteration, samp, s, v, u;
for(iteration=0; iteration<MAXITERATION; iteration++){
   x.error=0;
   for(samp=0; samp<SET; samp++){
   x = putInputsTrainSet(x,samp);
   x = propagation(x);
   x = error (x, samp);
   x.error += 0.5*x.error;
for(s=0; s<NUM_LAYERS-1; s++){
     for(v=0; v<x.layers[s+1].n; v++) {
       for(u=0; u<x.layers[s].n; u++) {
         x.weightP[s][u][v] = NI*x.layers[
s+1].sigma[v]*x.layers[s].output[u];
         x.weight[s][u][v] += x.weightP[s][u][v];
       }
x.layers[s+1].biasp[v] =
```

```
NI*x.layers[s+1].sigma[v];
        x.layers[s+1].bias[v] += x.layers[s+1].biasp[v];
}
}
}
if(x.error<MAXERROR)
    break;
  }
return x;
}
```

References

[1] http://www.sparkfun.com/datasheets/DevTools/PIC/PIC32MX3XX4XX.pdf.

[2] http://www.networksystemsdesignline.com/showArticle.jhtml;
jsessionid=HYWPNQQUKTOYLQE1GHOSKHWATMY32JVN?
articleID=162100251&queryText=pulse+width+modulation.

[3] D. L. Fried. Aperture averaging of scintillation. *J. Opt. Soc. Amer.*, 57:169–175, 1967.

[4] The Math Works Inc. Neural Network Toolbox, 1994.

[5] P. Stroud. Statistics of intermediate duration averages of atmospheric scintillation. *Opt. Eng.*, 35:543–548, 1996.

[6] M. Toyoshima and K. Araki. Effects of time averaging on optical scintillation in a ground-to-satellite atmospheric propagation. *Applied Optics*, 39:1911–1919, 2000.